

# Computación Inteligente: Grandes Modelos de Lenguajes

## Clase 5: Ingeniería de *Prompts*

Dr. Wladimir Rodríguez

wladimir@ula.ve

Profesor Titular

Escuela de Ingeniería de Sistemas

ULA

# Ingeniería de *Prompts*

- Introducción
- Técnicas Fundamentales
- Técnicas Avanzadas
- Evaluación y Métricas
- Herramientas y Recursos
- Ética y Consideraciones Sociales

# ¿Qué es la Ingeniería de *Prompts*?

- La ingeniería de *prompts* es el arte y la ciencia de diseñar y optimizar instrucciones o "prompts" para obtener los mejores resultados posibles de los modelos de lenguaje y otros sistemas de Inteligencia Artificial. Este campo combina elementos de psicología, lingüística, programación y diseño de interacción para crear comunicaciones efectivas con sistemas de Inteligencia Artificial.

# ¿Por qué es Importante la Ingeniería de *Prompts*?

- **Calidad de la Respuesta:** Un buen prompt = Respuestas más precisas, relevantes y útiles.
- **Control del LLM:** Dirige el modelo para evitar respuestas no deseadas (sesgos, alucinaciones, etc.).
- **Eficiencia:** Ahorra tiempo y recursos al obtener resultados óptimos desde el principio.
- **Creatividad:** Explora las capacidades del LLM para generar contenido innovador y sorprendente.

# Conceptos Fundamentales

- **Token:** La unidad básica de procesamiento del LLM (palabra, subpalabra, puntuación).  
(Ejemplo: "El gato negro" = 3 tokens)
- **Contexto:** La ventana de texto que el LLM considera al generar la respuesta.  
(Importancia de mantener el contexto relevante)
- **Temperatura:** Controla la aleatoriedad de la respuesta (0 = determinista, 2 = aleatorio).
- **Top P:** Limita las opciones de tokens más probables para generar la respuesta.
- **Penalidad:** Reduce la probabilidad de que el modelo repita palabras o frases.

# Temperatura

- » **Temperatura Alta (e.g., 1.0 o superior):**

- » El modelo se vuelve más "creativo" y puede generar respuestas más sorprendentes o inusuales.
- » La salida puede ser menos coherente o lógica, ya que el modelo no se limita a las opciones más probables.
- » Es útil para tareas que requieren diversidad o creatividad, como la escritura de poemas, historias o ideas innovadoras.

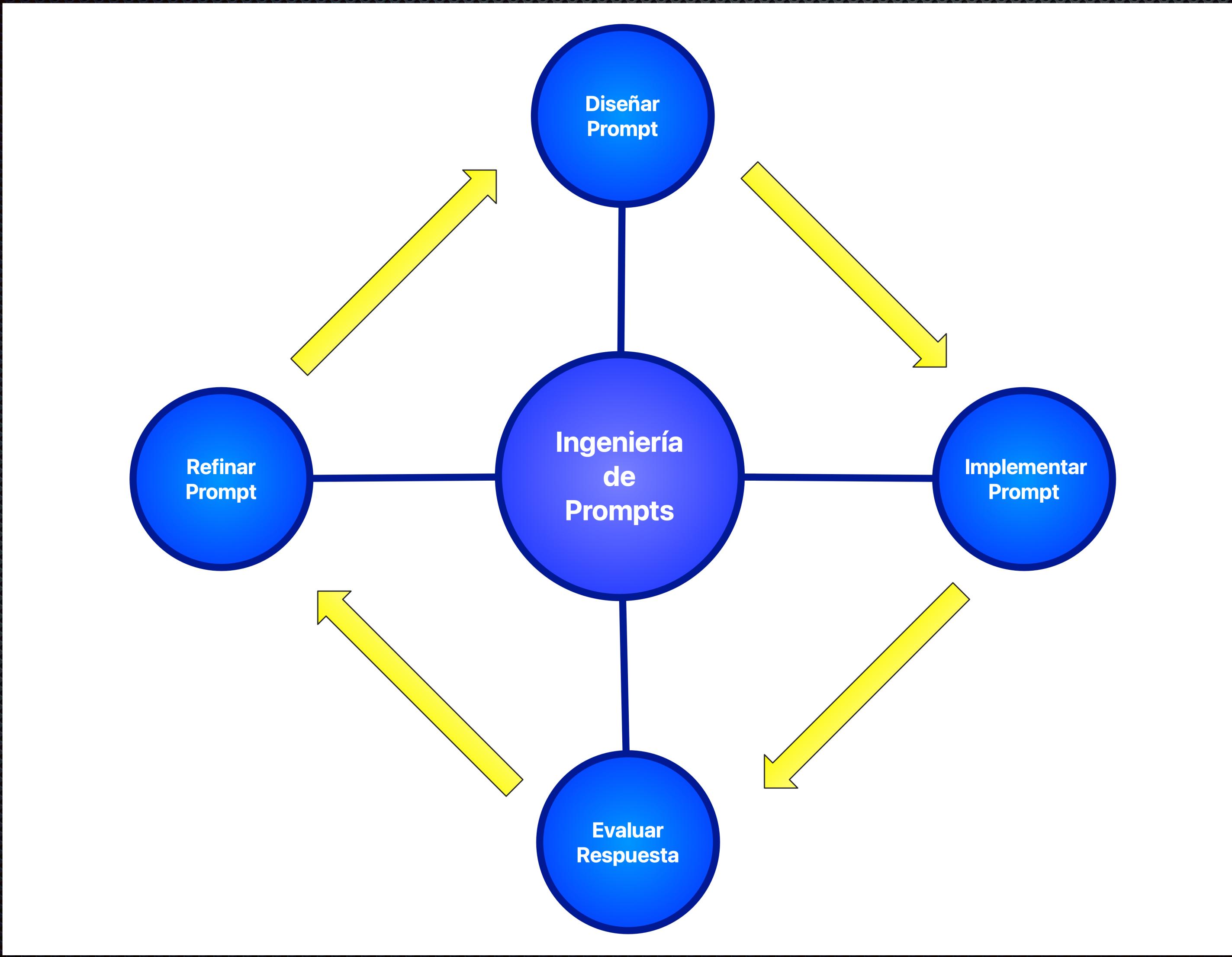
- » **Temperatura Baja (e.g., 0.5 o inferior):**

- » El modelo se vuelve más "lógico" y predecible, seleccionando las palabras más probables en cada paso.
- » La salida es más coherente y estructurada, lo que la hace más adecuada para tareas que requieren precisión, como la traducción o la contestación de preguntas concretas.
- » Sin embargo, puede resultar en respuestas demasiado repetitivas o previsibles.

# Top-p

- Cuando un LLM genera texto, para cada token tiene una distribución de probabilidades de las siguientes palabras posibles
- Top-p selecciona el subconjunto más pequeño de palabras cuyas probabilidades acumuladas suman el valor p
- Solo se consideran las palabras de este subconjunto para la selección final
- Por ejemplo, si top-p = 0.9:
  - El modelo ordena todas las palabras posibles por probabilidad
  - Suma las probabilidades empezando por la más alta
  - Se detiene cuando la suma alcanza 0.9 (90%)
  - Solo considera este grupo de palabras para la selección final

# El Ciclo de Vida del Prompt



# El Ciclo de Vida del *Prompt*

## » Diseño:

- **Análisis de la tarea:** comprender claramente el objetivo que se quiere lograr con el prompt.
- **Investigación:** explorar diferentes enfoques y técnicas de prompting.
- **Creación del prompt inicial:** formular un prompt que sea claro, conciso y específico.

## » Implementación:

- **Selección del LLM:** elegir el modelo más adecuado para la tarea.
- **Configuración de parámetros:** ajustar los parámetros de generación (temperatura, top\_p, etc.) para obtener los resultados deseados.
- **Ejecución del prompt:** enviar el prompt al LLM y obtener la respuesta.

# El Ciclo de Vida del *Prompt*

- **Evaluación:**
  - **Análisis de la respuesta:** evaluar la calidad, la relevancia y la precisión de la respuesta.
  - **Identificación de problemas:** detectar posibles sesgos, alucinaciones o errores.
  - **Medición del rendimiento:** utilizar métricas objetivas y subjetivas para evaluar el rendimiento del prompt.
- **Refinamiento:**
  - **Iteración:** modificar el prompt basándose en los resultados de la evaluación.
  - **Experimentación:** probar diferentes variaciones del prompt para encontrar la mejor solución.
  - **Optimización:** ajustar los parámetros de generación para mejorar el rendimiento.

# Elementos Clave de un *Prompt*

- **Instrucción.** Esta es la directriz principal del *prompt*. Indica al modelo lo que quieras que haga. Por ejemplo, "Resume el siguiente texto" proporciona una acción clara para el modelo.
- **Contexto.** El contexto proporciona información adicional que ayuda al modelo a comprender la perspectiva general o los antecedentes. Por ejemplo, "Teniendo en cuenta la recesión económica, proporciona asesoramiento de inversión" da al modelo un telón de fondo en el que enmarcar su respuesta.
- **Datos de entrada.** Es la información o los datos concretos que quieras que procese el modelo. Puede ser un párrafo, un conjunto de números o incluso una sola palabra.
- **Indicador de salida.** Especialmente útil en representaciones de roles, este elemento orienta al modelo sobre el formato o tipo de respuesta deseada. Por ejemplo, "Reescribe la siguiente frase con el estilo de Shakespeare" da al modelo una instrucción estilística.

# Tipos de Prompts

- » Preguntas:
  - » Preguntas directas: "¿Cuál es la capital de Francia?"
  - » Preguntas abiertas: "Describe la Revolución Francesa."
  - » Preguntas hipotéticas: "¿Qué pasaría si la Revolución Francesa nunca hubiera ocurrido?"
- » Instrucciones:
  - » Instrucciones simples: "Escribe un resumen de este artículo."
  - » Instrucciones detalladas: "Escribe un resumen de este artículo, enfocándote en los aspectos económicos y sociales, y utilizando un lenguaje claro y conciso."
  - » Instrucciones con restricciones: "Escribe un poema de 10 versos sobre el amor, utilizando la rima consonante."

# Técnicas Básicas de Ingeniería de *Prompt*

- *Prompt* sin entrenamiento previo (Zero-Shot Prompting)
- *Prompt* con pocos ejemplos (Few-Shot Prompting)
- Prompt Cadena de Pensamiento (Chain-of-Thought Prompting)
- Role Prompting

# *Prompt* sin entrenamiento previo

- **Aprovechar el Conocimiento Pre-Existente del Modelo:**
  - **La Idea Central:** *Prompt* sin entrenamiento previo se basa en la capacidad de los LLMs para generalizar y realizar tareas sin haber sido explícitamente entrenados para ellas. Se aprovecha el conocimiento que el modelo ya ha adquirido durante su pre-entrenamiento masivo.
  - **Confiar en la Generalización:** La clave es formular prompts que activen el conocimiento relevante que el modelo ya posee. Esto requiere comprender qué tipo de información y habilidades se espera que el modelo haya aprendido.
  - **Ejemplo:** Un LLM pre-entrenado en una gran cantidad de texto probablemente tendrá conocimiento sobre la capital de Francia, incluso si nunca fue entrenado explícitamente para responder a esa pregunta.

# *Prompt* sin entrenamiento previo

- **Formulación de Preguntas Directas:**
  - **Claridad es Clave:** La pregunta debe ser clara, concisa y directa. Evitar ambigüedades y lenguaje complicado.
  - **Contexto Mínimo:** En general, el Prompt sin entrenamiento previo funciona mejor cuando se proporciona el contexto mínimo necesario para que el modelo comprenda la pregunta. Demasiado contexto puede confundir al modelo.
- **Ejemplos:**
  - "Traduce 'Hola mundo' al inglés."
  - "¿Cuál es la capital de Italia?"
  - "Resume el siguiente texto: [Texto]."

# *Prompt* sin entrenamiento previo

- **Limitaciones y Casos de Uso:**
  - **Dependencia del Conocimiento Pre-Existente:** El Prompt sin entrenamiento previo solo funciona si el modelo ya tiene el conocimiento o la habilidad necesaria para realizar la tarea.
  - **Rendimiento Variable:** El rendimiento puede variar significativamente dependiendo de la tarea y del modelo utilizado.
  - **Dificultad con Tareas Complejas:** El Prompt sin entrenamiento previo suele ser menos efectivo para tareas complejas que requieren razonamiento, planificación o conocimiento especializado.
  - **Casos de Uso:**
    - Tareas simples de clasificación de texto (e.g., análisis de sentimiento).
    - Traducción automática básica.
    - Generación de texto simple.

# *Prompt* con pocos ejemplos

- **Proporcionar Ejemplos para Guiar al Modelo:**
  - **La Idea Central:** *Prompt* con pocos ejemplos consiste en proporcionar al LLM algunos ejemplos de entrada y salida para guiarlo en la realización de una tarea específica. Esto ayuda al modelo a comprender el patrón que debe seguir y a generalizar a partir de los ejemplos proporcionados.
  - **Aprendizaje en Contexto:** El LLM aprende a realizar la tarea "en contexto", sin necesidad de ser explícitamente entrenado para ella.
  - **Mejora Significativa:** *Prompt* con pocos ejemplos suele mejorar significativamente el rendimiento en comparación con zero-shot prompting, especialmente para tareas complejas.

# *Prompt con pocos ejemplos*

- **Selección de Ejemplos Relevantes y Diversos:**
  - **Relevancia:** Los ejemplos deben ser relevantes para la tarea que se quiere realizar. Deben representar el tipo de entrada y salida que se espera del modelo.
  - **Diversidad:** Los ejemplos deben ser diversos, cubriendo diferentes aspectos de la tarea y evitando la redundancia. Esto ayuda al modelo a generalizar mejor.
  - **Calidad:** Los ejemplos deben ser de alta calidad, sin errores ni ambigüedades.
  - **Número de Ejemplos:** El número óptimo de ejemplos depende de la tarea y del modelo utilizado. En general, más ejemplos conducen a un mejor rendimiento, pero hay un punto de rendimientos decrecientes

# *Prompt* con pocos ejemplos

- **Formato y Estructura de los Ejemplos:**
  - **Consistencia:** Los ejemplos deben tener un formato consistente, con una estructura clara y fácil de entender.
  - **Delimitadores:** Utilizar delimitadores claros (e.g., "Entrada:", "Salida:") para separar la entrada de la salida.
  - **Ejemplo:**
    - **Prompt = la: 2, mas: 3, casa: 4, nuevo:**
    - **Salida = 5**

# *Prompt Cadena de Pensamiento*

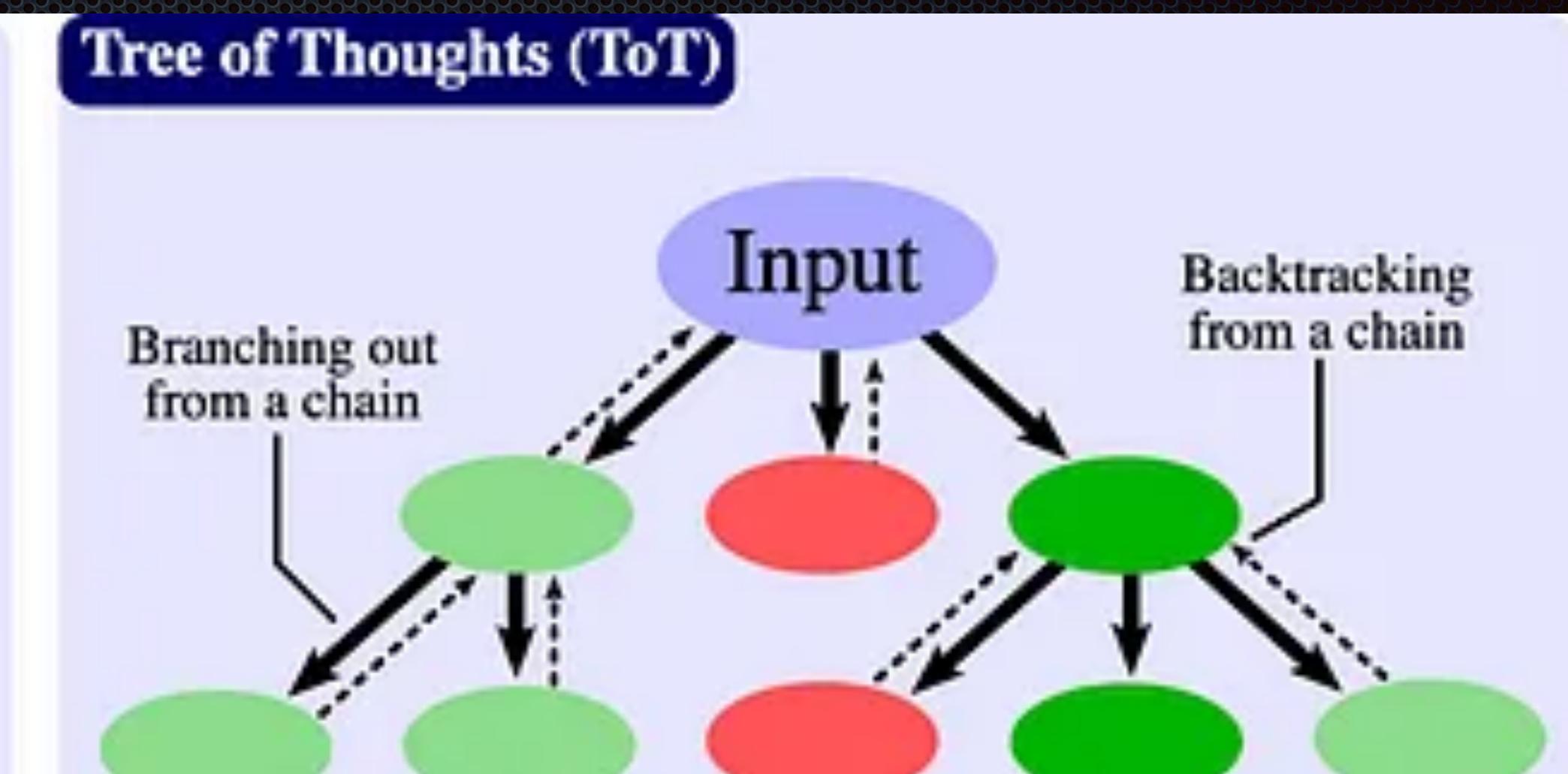
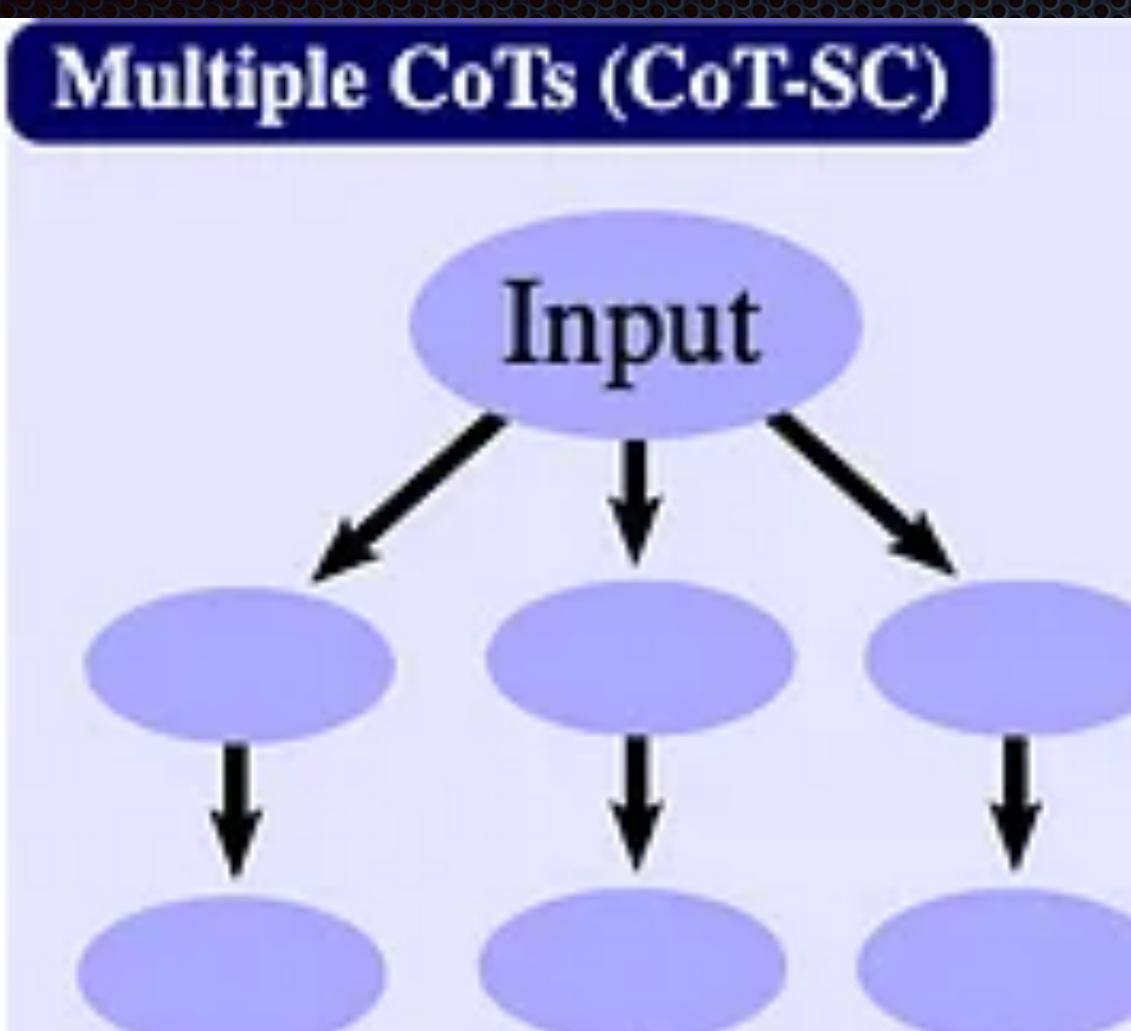
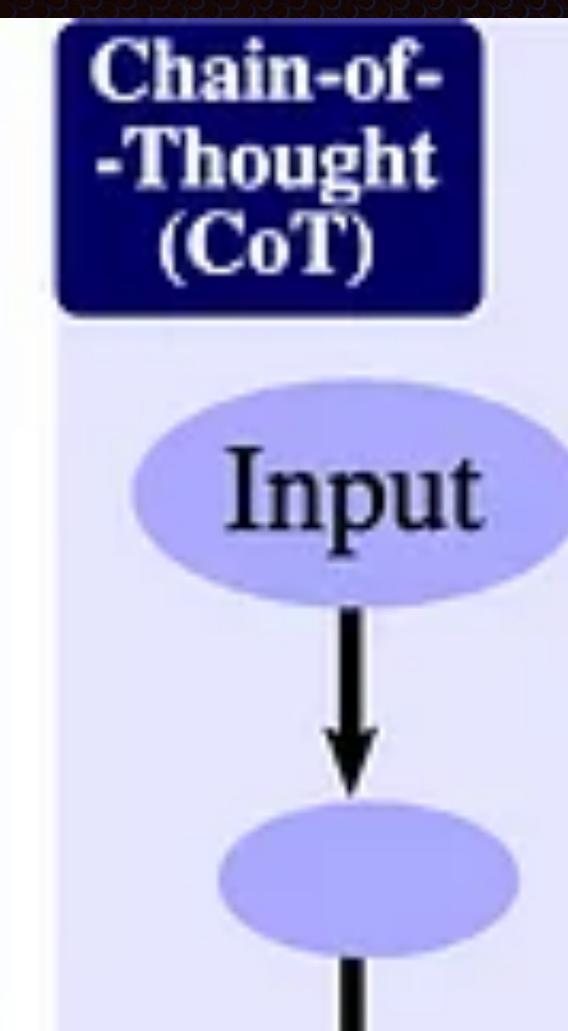
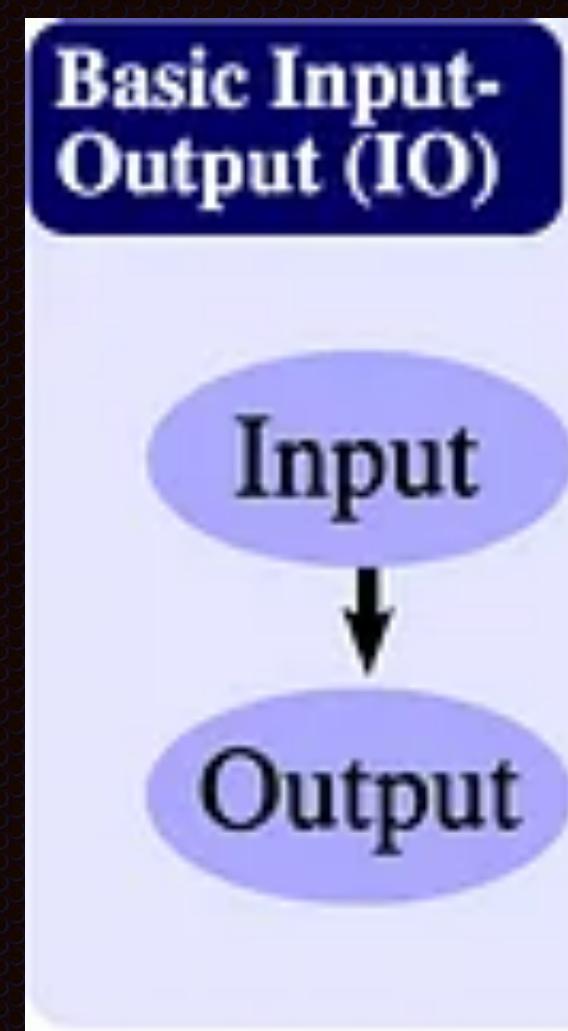
- **Fomentar el Razonamiento Paso a Paso del Modelo:**
  - **La Idea Central:** Prompt Cadena de Pensamiento (CoT) es una técnica que consiste en guiar al LLM para que rzone paso a paso a través de un problema, en lugar de simplemente proporcionar la respuesta final. Esto mejora la precisión y la explicabilidad de las respuestas.
  - **Imitar el Pensamiento Humano:** Se busca que el LLM imite el proceso de pensamiento humano, descomponiendo el problema en pasos más pequeños y resolviendo cada paso de forma lógica.
  - **Mejora en Tareas de Razonamiento Complejas:** CoT prompting es especialmente útil para tareas que requieren razonamiento matemático, lógico o de sentido común.

# *Prompt Cadena de Pensamiento*

- “Pensemos paso a paso” y Otras Frases Clave:
  - **La Frase Mágica:** La frase "Pensemos paso a paso" ha demostrado ser muy efectiva para activar el razonamiento en cadena en los LLMs.
  - **Otras Frases:** Otras frases útiles incluyen "Primero, necesitamos...", "Luego, calculamos...", "Finalmente, la respuesta es...".
- **Ejemplo:**
  - **Pregunta:** Juan tiene 5 manzanas. María le da 3 manzanas más. ¿Cuántas manzanas tiene Juan en total? Pensemos paso a paso.

# *Prompt Cadena de Pensamiento*

- **Mejora en la Precisión y la Explicación de las Respuestas:**
  - **Mayor Precisión:** CoT prompting ayuda a reducir los errores y a mejorar la precisión de las respuestas, especialmente en tareas complejas.
  - **Explicabilidad:** CoT prompting proporciona una explicación clara y concisa del proceso de razonamiento del LLM, lo que facilita la comprensión y la verificación de la respuesta.
  - **Ejemplo:** En lugar de simplemente dar la respuesta "8", el LLM explicará que primero sumó las 5 manzanas de Juan con las 3 manzanas de María para obtener el total de 8 manzanas.



# *Role Prompting*

- El role prompting es una técnica utilizada en el ámbito de los modelos de lenguaje (LLMs, por sus siglas en inglés) que consiste en asignar un rol específico al modelo para guiar su comportamiento y mejorar la calidad de sus respuestas. Al darle un rol al modelo, se le proporciona un contexto claro sobre cómo debe actuar, qué tono usar y qué tipo de información debe priorizar. Esto es especialmente útil para obtener respuestas más coherentes, precisas y adaptadas a un contexto particular.

# ¿Cómo Funciona el *Role Prompting*?

- **Asignación de un rol:** Se le indica al modelo que adopte un rol específico, como "experto en historia", "asesor financiero", "profesor de matemáticas", "escritor creativo", etc.
- **Contextualización:** El rol define el tono, el estilo y el enfoque que el modelo debe seguir al generar respuestas.
- **Mejora de la calidad:** Al adoptar un rol, el modelo tiende a generar respuestas más especializadas y adaptadas a las necesidades del usuario.

# Ejemplos de *Role Prompting*

- **Rol de experto en historia:**

- **Prompt:** "Eres un historiador especializado en la Edad Media. Explícame las causas de la caída del Imperio Romano."
- **Respuesta:** El modelo generará una explicación detallada y precisa, utilizando un tono académico y basándose en hechos históricos.

- **Rol de asesor financiero:**

- **Prompt:** "Eres un asesor financiero profesional. Recomiéndame estrategias de inversión para alguien que quiere empezar con un capital pequeño."
- **Respuesta:** El modelo proporcionará consejos prácticos y realistas, utilizando un lenguaje profesional y enfocado en finanzas.

# Ejemplos de *Role Prompting*

- **Ejemplo práctico**
- **Prompt sin role prompting:**
  - "Explícame cómo funciona la fotosíntesis."
- **Prompt con role prompting:**
  - "Eres un biólogo especializado en botánica. Explícame de manera sencilla cómo funciona la fotosíntesis para un público sin conocimientos científicos."

# Consejos para usar *Role Prompting*

- **Define el rol claramente:** Cuanto más específico sea el rol, mejor será la respuesta.
- **Proporciona contexto adicional:** Si es necesario, incluye detalles sobre el tema o el público objetivo.
- **Experimenta con diferentes roles:** Prueba distintos roles para ver cuál se adapta mejor a tus necesidades.
- **Combina con otras técnicas:** Puedes usar role prompting junto con otras técnicas como chain-of-thought o few-shot learning para mejorar aún más los resultados.

# Beneficios del *Role Prompting*

- **Precisión:** El modelo se enfoca en generar respuestas más específicas y relevantes.
- **Coherencia:** El rol ayuda a mantener un tono y estilo consistentes.
- **Adaptabilidad:** Permite personalizar las respuestas según el contexto o la audiencia.
- **Eficiencia:** Reduce la necesidad de ajustar manualmente el prompt para obtener respuestas útiles.

# Implicaciones Sociales de los LLM

- **Desinformación y noticias falsas:** Los LLM se pueden utilizar para generar noticias falsas y desinformación de manera rápida y convincente. Esto puede tener un impacto significativo en la opinión pública y el discurso político.
- **Pérdida de empleos:** A medida que los LLM se vuelven más avanzados, existe la preocupación de que puedan automatizar muchos trabajos que actualmente realizan los humanos. Esto podría conducir a la pérdida de empleos y al aumento de la desigualdad.
- **Sesgo y discriminación:** Los LLM se entrena en grandes cantidades de datos, que pueden contener sesgos. Esto puede llevar a que los LLM perpetúen o incluso amplifiquen los sesgos existentes en la sociedad.
- **Dependencia tecnológica:** A medida que dependemos más de los LLM para realizar tareas, existe el riesgo de que nos volvamos demasiado dependientes de la tecnología. Esto podría tener un impacto negativo en nuestras propias habilidades y capacidades.

# Implicaciones Éticas de los LLM

- **Responsabilidad:** Es importante determinar quién es responsable de las acciones de un LLM. Si un LLM causa daño, ¿quién es responsable?
- **Transparencia:** Los LLM pueden ser difíciles de entender y explicar. Esto puede dificultar la determinación de si un LLM está actuando de manera ética.
- **Privacidad:** Los LLM pueden recopilar y almacenar grandes cantidades de datos personales. Es importante proteger la privacidad de las personas y garantizar que sus datos se utilicen de manera responsable.
- **Control:** Es importante garantizar que los humanos tengan el control de los LLM. No debemos permitir que los LLM tomen decisiones que tengan un impacto significativo en nuestras vidas sin la supervisión humana.